Fermi National Accelerator Laboratory

FERMILAB-Conf-98/360

Processing Farms Plans for CDF and D0 for Run II

M. Breitung et al.

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, Illinois 60510

December 1998

Published Proceedings of *Computing in High Energy and Nuclear Physics (CHEP)*, Chicago, Illinois, August 31-September 4, 1998

Operated by Universities Research Association Inc. under Contract No. DE-AC02-76CH03000 with the United States Department of Energy

Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Distribution

Approved for public release; further dissemination unlimited.

Copyright Notification

Processing Farms Plans for CDF and D0 for Run II

Mark Breitung, Liz Buckley-Geer, Yen-Chu Chen, Mike Diesburg,
Phil DeMar, Jim Fromm, Don Holmgren, Stephan Lammel, Lee Lueking,
Tanya Levshina, Igor Mandrichenko, Heidi Schellman,
Marilyn Schweitzer, Dane Skow, Steve Wolbers, G. P. Yeh*

Fermi National Accelerator Laboratory, Batavia, IL

CDF and D0 computing needs for the next collider run (Run II) are roughly 30 times that of those used during Run I. Then, Farms of approximately 3000 total MIPS (100 UNIX workstations) processed the 100 million recorded events. Run II processing may require over 100,000 MIPS. This increase requires large and robust hardware and software systems for smooth processing. Described are the proposed Run II Farm architecture and plans to ensure its suitability.

1 Introduction

The driving force behind the Farms architecture is to provide sufficient CPU capacity for CDF and D0 reconstruction. Extra CPU must also be provided for Monte Carlo, Stripping and Reprocessing. These estimates are large, so it is especially important to maximize the CPU efficiency and keep costs down. Eliminating outmoded and high maintenance technologies, introducing new cost effective technologies, and providing systems that run efficiently 24 hours a day and 7 days a week are some ways to achieve this.

A Central Mass Storage System has a key role overall in Run II and the Farms must pay particular attention the CDF and D0 data access techniques to it.

Deployment must be completed in time for the following milestones:

• 1st Phase of Farms Purchase	Spring, 1999
• CDF/D0 Mock Data Challenge	Fall, 1999
• First Collisions	April, 2000

2 CPU Requirements

CDF and D0 provide estimates in terms of a Fermilab MIP which is approximately equal to 7 hundredths of a SPECint95 MIP or approximately 3 SPECint92 MIPs. Table 1 shows the estimated MIPS required to handle the DC rate for event reconstruction alone.

Element		**CDF**		**D0**	
		Min.	**Max.**	**Min.**	**Max.**
Raw Event Size (KBytes)		250		250	
Events/Second	**Peak Hz**	75		50	
	DC Hz	28		20	
MIPs/Event		1200	1800	2000	5000
MIPs to keep up with DC	**at 100% efficiency**	33,600	50,400	40,000	100,000
	at 70% efficiency	48,000	72,000	58,000	143,000

Table 1 Estimated MIPS for Reconstruction

Table 2 shows the CDF and D0 requirements considering all types of processing anticipated to be handled on the farms:

Type of Processing		**Estimated MIPs**			
		CDF		**D0**	
		Min.	**Max.**	**Min.**	**Max.**
Reconstruction for DC at 70% efficiency		48,000	72,000	58,000	143,000
Monte Carlo		10,000	40,000	20,000	40,000
Reprocessing		24,000	24,000	30,000	70,000
Stripping		10,000	10,000	-	
Total (processing types overlaps considered)	**by year 2000**	50,000	70,000	80,000	160,000
	by year 2001	72,000	96,000	110,000	253,000

Table 2 Estimated MIPS For All Types Of Processing

Measurements show that a 200MHz PC provides 115 MIPs. Extrapolation implies that a 400MHz PC can deliver 230 MIPs and a 500MHz PC can deliver 287 MIPs. Thus, the number of nodes required for Run II can be estimated as shown in Table 3:

Estimated MIPs Considering Overlaps In Processing Types			**Number of Dual Processor PCs**	
			400 MHz	**500 MHz**
Total year 2000	**CDF**	50,000 - 70,000	110 - 150	90 - 125
	D0	80,000 - 160,000	175 - 350	140 - 280
	CDF+D0	130,000 - 230,000	285 - 500	230 - 405
Total year 2001	**CDF**	72,000 - 96,000	160 - 210	125 - 170
	D0	110,000 - 253,000	240 - 550	195 - 440
	CDF+D0	182,000 - 349,000	400 - 760	320 - 610

Table 3 Estimated Number of PCs

Though rather daunting in scope, Fermilab has successfully managed Farms of over 300 nodes before.

3 Overall Architectural Decisions

Large arrays of PCs running the Linux operating system are prime candidates for Run II computing because of their low cost and experience with PCs has so far been very favorable. Linux may also offer more control for administrators over managing multiple versions of commercial UNIX operating systems. But, PCs may require more effort in terms of purchasing, system integration, and overall operations. Other technologies (e.g. commercial UNIX workstations or SMPs) may be preferred for file servers or centralized control systems and are not ruled out until more experience with PCs is gained.

Previously, the Farms relied heavily on locally attached tape drives for I/O. While cost effective in the past, they are overall a high maintenance item. With the advent of the Central Mass Storage System, it should be more efficient and cost effective to go through the Mass Storage System for data and eliminate locally attached tape drives on the Farms.

Logically, nodes will be classified into two primary types:

- Job Manager nodes that allow full user access, job submission and a central control point for user jobs

- Execution nodes that are dedicated to computational and/or I/O tasks and have relatively restricted interactive user access.

The Farms Batch System architecture should be flexible enough to allow different ways of logically dividing nodes into types. The actual configuration of nodes is dictated by hardware capabilities and the volume and nature of computations performed by end users. The batch system should allow fast, flexible and non-intrusive tuning of its configuration.

LSF, a commercial batch system developed by Platform Computing may be beneficial as the core of the Farms Batch System. Fermilab has used LSF successfully for several years and using it will help minimize of the number of supported batch architectures.

To simplify CDF and D0 reconstruction software, file based control software is preferred over event based software. Thus, Cooperative Processes Software (CPS) will not be used as it was for Run I and Fixed Target experiments.

4 Hardware Architecture

Two hardware models are being investigated. One considers all nodes to be identically configured with adequate disk and direct access to the Mass Storage System. The other (see FIGURE 1.) considers the two types of nodes:

- a few I/0 nodes that have large disk space, fast/direct access to the Mass Storage System, and are dedicated primarily to I/O-bound tasks.
- many worker nodes that have limited disk space, get data for processing from an I/O node, and are dedicated primarily to CPU-bound tasks.

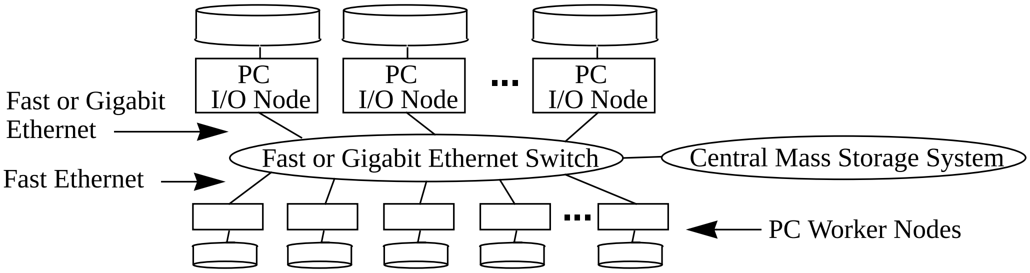


FIGURE 1. Hardware Architecture with I/0 and CPU Nodes differentiated

The advantages and disadvantages of each model are being investigated based upon:

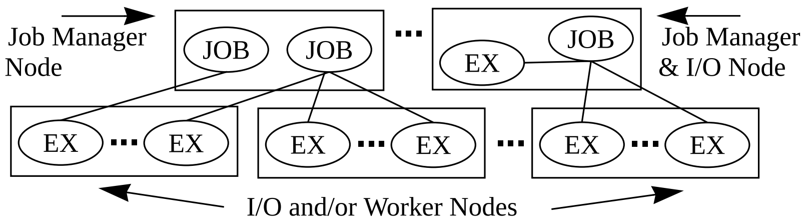
- Ease of maximizing and managing efficient CPU utilization and I/O performance
- Ease of managing data flow to the Mass Storage System
- Dedicated versus shared data and interactive network traffic
- Complexity of over all job control
- Management and cost of distributed versus centralized disk

In terms of network fabric, Ethernet (fast and/or gigabit) is preferred so far over Fibre Channel Standard based on its lack of complexity and low cost per port.

A Prototype (see Section 7 on page 6) will be used to finalize the hardware architecture.

5 Software Architecture

The major software element of the Run II Production Management software is a batch system that works in concert with the Run II CDF and D0 Data Access Methods where users executables are dispatched to physical nodes in the Farms Batch System (see FIGURE 2.).



JOB: A user job which specifies one or more CPU intensive tasks and/or I/O tasks to be remotely executed

EX: A CPU intensive executable or an I/O executable

FIGURE 2. Proposed Overall Software Architecture

The Farms Batch System must provide the following functionality:

- Job Control for software (e.g. queue jobs, execute user processes, report job/process status, coordinate job steps, notify user upon job when completion)
- Resource Management (e.g. node status, buffer/scratch disk status, authorized users, number of allowable executables per node)
- Management Tools (e.g. job submission/cancellation, job hold/release, job monitoring, job history /statistics, farm shutdown/start-up)

The Farm Batch System is proposed to consist of the following software elements:

- LSF for high level batch control on Job Manager nodes including ELIM interfaces that monitors Farm node and Farm scratch space availability.
- Farm Load Information Manager Daemon (FLIMD) for central management of nodes
- Job Manager (JM) to control and monitor a particular section of Farm job
- Farm Daemon (FARMD) to start, monitor and control user processes on a node
- Farms Data Storage (FDS) for a scratch disk space management
- User Interface for job submission, monitoring, status, history, administration, etc...
- Configuration file for the Farms configuration (e.g. node name and type)
- Farms Logging Daemon (FLOGD) to centrally log all error and output messages
- a Job Description File (JDF) consisting Job Sections describing a user’s Farm job.

When a Farm job is submitted, the JDF file (see FIGURE 3.) is processed and an LSF job submitted for each Job Section specifying the JM as the executable (see FIGURE 4.).

```
SECTION in_stage# section name
    EXEC = ”dump.exe”                    # executable location & name
    QUEUE = test_io                      # queue name
    NUMPROC = 1                          # number of processes
    STDOUT = “/usr/home/joe/out/in”      # location for stdout & stderr
    STDERR = “/usr/home/joe/err/in”      # - defaults to $HOME
SECTION reco
    EXEC = ”worker.exe arg1 arg2 ...”
    NUMPROC = 10
    QUEUE = test_worker
    STDOUT = “/usr/home/joe/out/reco”
    STDERR = “/usr/home/joe/err/reco”
    DEPEND = done(in_stage)              # started(), done(), exit(),or finished()
SECTION out_stage
    EXEC = ”/usr/home/joe/collect.exe”
    NUMPROC = 1
    DEPEND  = done(reco)
    QUEUE = test_io
    STDOUT = “/usr/home/joe/out/out”
    STDERR = “/usr/home/joe/err/out”
```

FIGURE 3. A Sample Job Description File

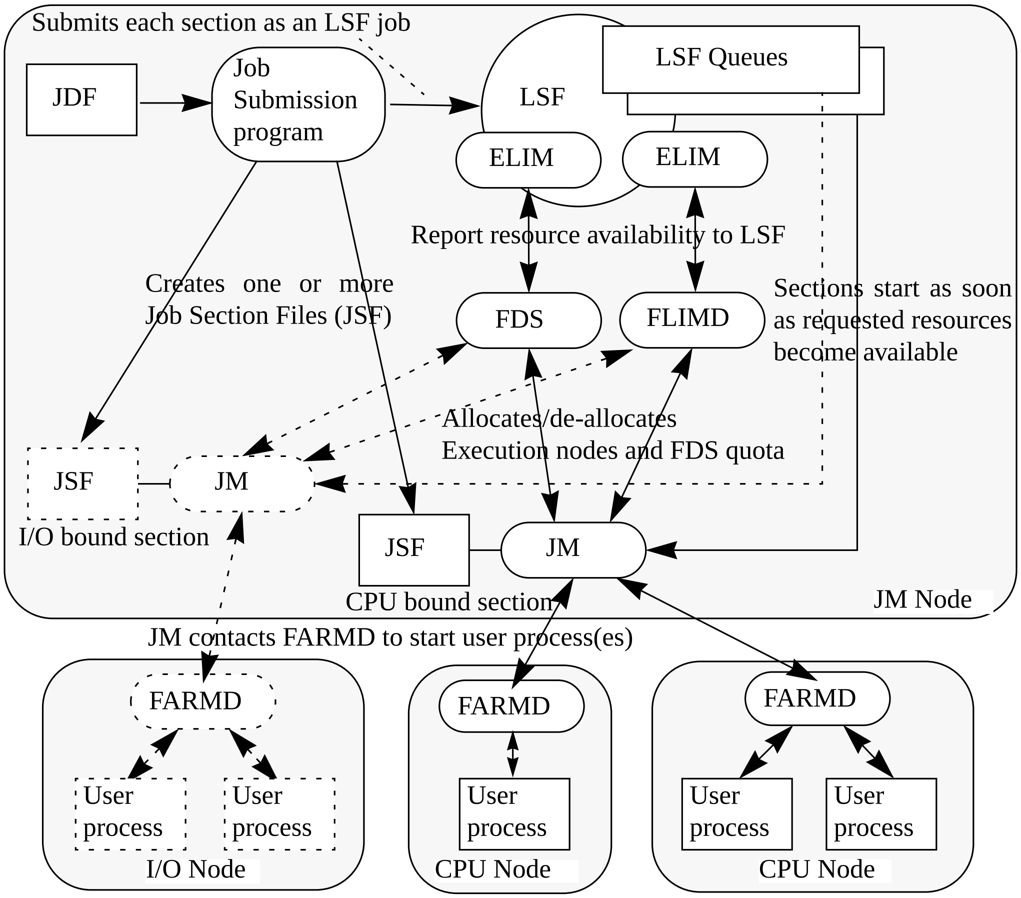

Submits each section as an LSF job
JDF
Job Submission program
LSF
LSF Queues
ELIM
ELIM
Report resource availability to LSF
Creates one or more Job Section Files (JSF)
FDS
FLIMD
Sections start as soon as requested resources become available
Allocates/de-allocates Execution nodes and FDS quota
JSF
JM
I/O bound section
JSF
JM
CPU bound section
JM Node
JM contacts FARMD to start user process(es)
FARMD
User process
User process
I/O Node
FARMD
User process
CPU Node
FARMD
User process
User process
CPU Node

FIGURE 4. Farm Batch System Architecture

6 Proof of Concept - Fixed Target PC Farm

Several events resulted in shortage of computing resources for the Fixed Target 1997 experiments. This gave us an opportunity to purchase, deploy, and administer a large number of PCs in a production environment before Run II, rather than simply expanding the existing IBM and SGI farms. Twenty-eight dual and eight single processor 333MHz Pentium PCs were purchased in July and will go into full production in October (See FIGURE 5.).

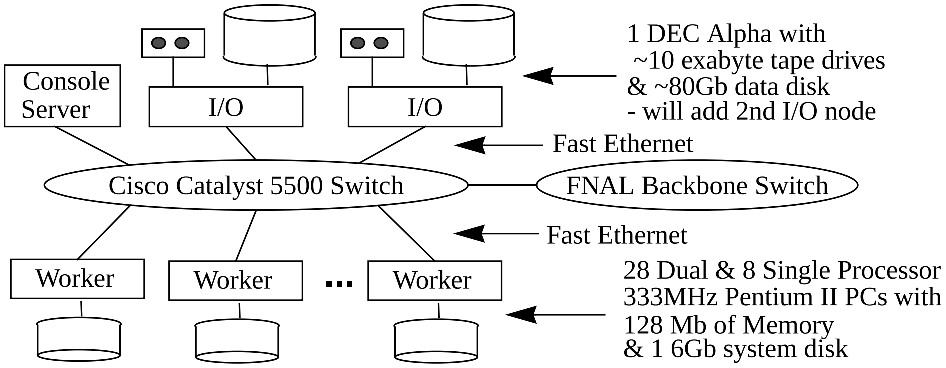


FIGURE 5. Fixed Target PC Farm

The original software plan for these PCs was to use CPS and the CPS batch system as is used on the IBM and SGI Fixed Target Farms. However, the Run II batch system prototype is used instead due to resulting complexities of CPS in a mixed mode architecture (Linux/OSF1) and the experiment's willingness to work with a software prototype. In this model, data is staged from tape to central I/O nodes, and read via NFS. The resulting output data is copied back to the I/O node, merged, and written to tape.

7 Run II and Theory Prototype Farm

A small scale farm was purchased for development and prototype work. CDF, D0 and the Computing Division will use it to provide a basis to finalize decisions on data flow, to gain experience with different switching fabrics, to provide a basis for the first phase of Run II hardware procurement, and to complete our study of our batch software model.

In addition, this prototype system will be used by theoretical physicists to study future architectures for their work. Though some of their processing requires very low latency systems and tightly coupled parallelization, other processing requirements are quite similar to CDF/D0 Run II requirements.

This prototype consists of four I/O nodes and fourteen worker nodes. Each node is a Dual processor 400MHz Pentium II PC. The I/O nodes differ from the worker nodes only in the amount of data disk provided. The prototype (see FIGURE 6.) provides plenty of flexibility to determine the final Run II hardware and networking configuration.

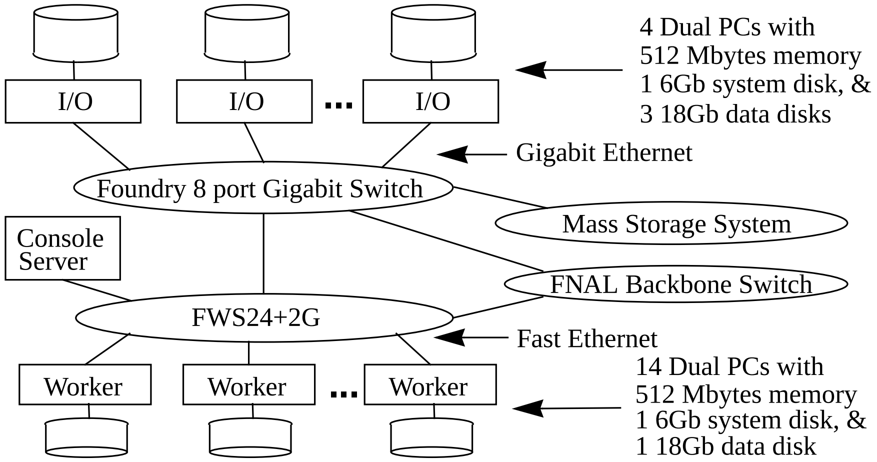


FIGURE 6. Run II and Theory Prototype Farm

8 Conclusions

Earlier investigations of PCs quite are quite promising. The Fixed Target PC Farm will complete our studies of using PCs in production and give us good early experience with our batch software model.

The Run II and Theory Prototype PC Farm will help us test new hardware, network switches, and software in a small, but very flexible configuration, thereby lowering the risks in future hardware acquisition for production.

When more accurate CDF and D0 CPU requirements become available, they will be used to help ensure that the hardware and software model scale well.

Finally, care must be taken to ensure that the feasibility studies are completed in time for the first phase of hardware procurement.